Bayesian anomaly detection

Samuel Alan Kossoff Leeney, Will Handley, Dominic Anstey, Eloy de Lera Acedo

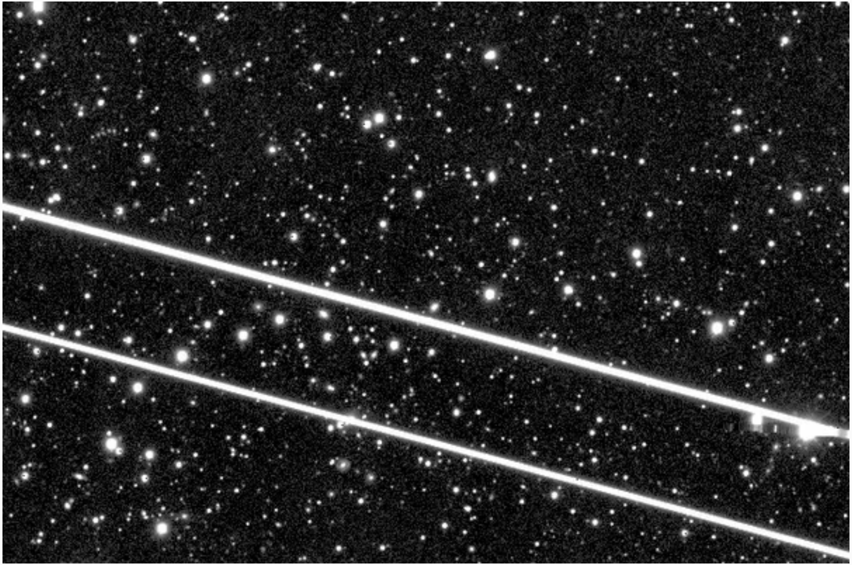
January 22, 2025

Will Handley's group meeting

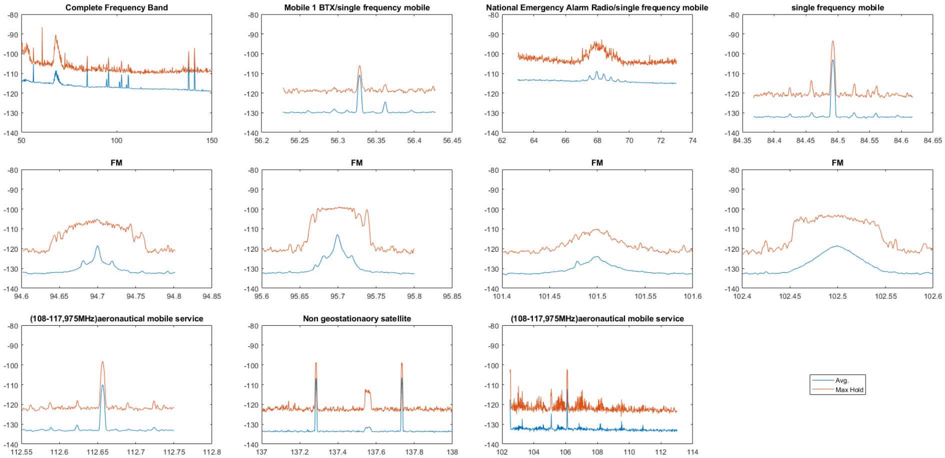
What is anomaly detection useful for?

1. **Contamination**
   - Simultaneously detecting and mitigating contaminated data.
2. **Anomalies**
   - Radio transients, cosmic ray flares, GW signals etc.)

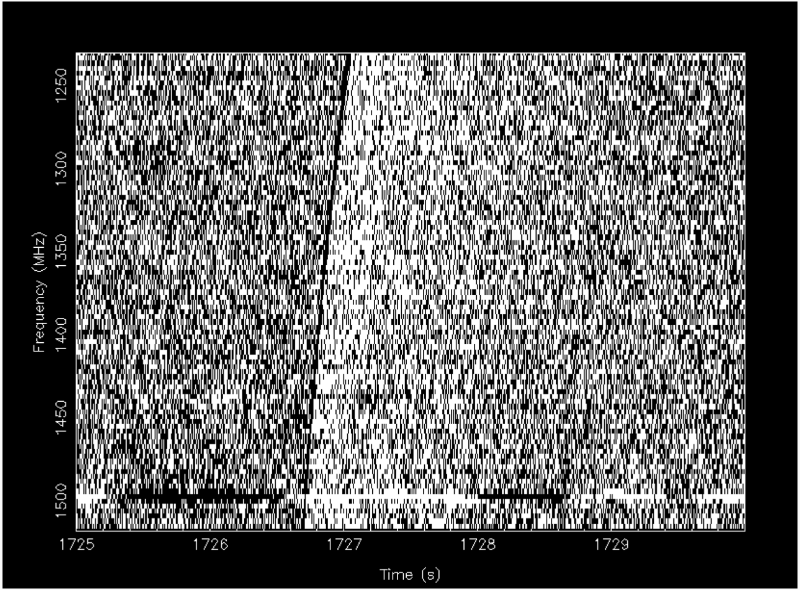
Contaminated data



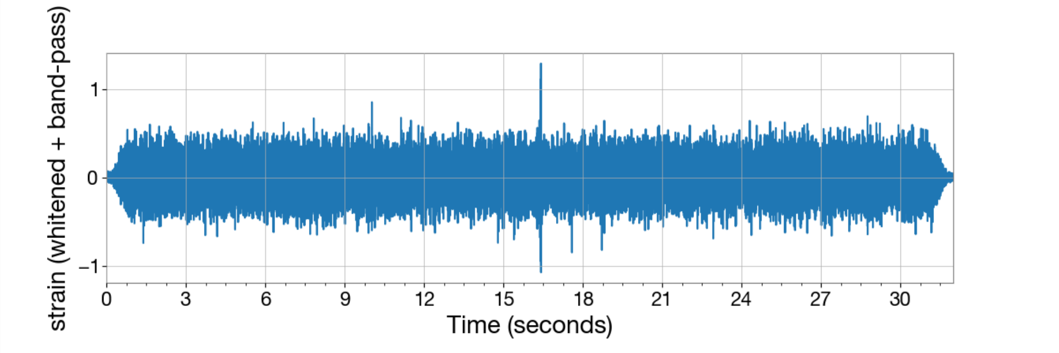
Contaminated data on the REACH site



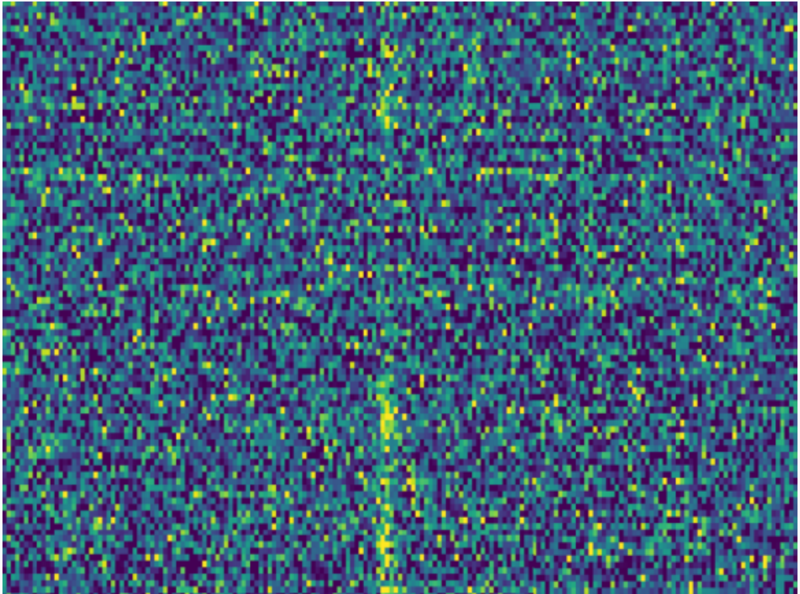
Anomalies - FRB 1



Anomalies (GW150914)



What do(nt) we look for?



Over simplified example of anomaly detection method (thresholding)

Simple statistical approach:

- Calculate mean (μ) and standard deviation (σ) of the data
- Define threshold $T = \mu + k\sigma$ where k is a sensitivity parameter
- Flag data point x_i as anomalous if:

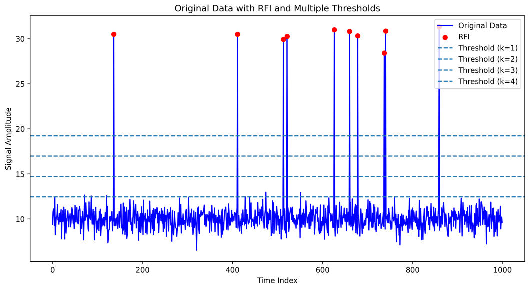
$$\text{anomaly}_i = \begin{cases} 1 & \text{if } x_i > T \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Limitations:

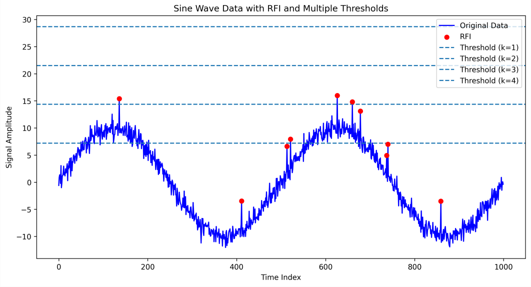
- Choice of k is arbitrary
- Assumes Gaussian statistics
- No consideration of temporal correlations
- Cannot distinguish between RFI and real signals

Thresholding results

Constant signal with RFI:



Sine wave with RFI:



Defining an anomaly sensitive likelihood

a) Generate peicewise likelihood:

$$P(\mathcal{D}_i|\theta) = \begin{cases} \mathcal{L}_i(\theta) & : \text{expected} \\ \Delta^{-1}[0 < \mathcal{D}_i < \Delta] & : \text{anomalous}, \end{cases} \tag{2}$$

b) Ascribe Bernoulli prior:

$$P(\varepsilon_i) = p_i^{(1-\varepsilon_i)}(1-p_i)^{\varepsilon_i}. \tag{3}$$

c) Marginalise over epsilon:

$$P(\mathcal{D}|\theta) = \sum_{\varepsilon\in\{0,1\}^N} P(\mathcal{D},\varepsilon|\theta) \tag{4}$$

d) Approximate correct mask is most likely

$$P(\mathcal{D}|\theta,\varepsilon_{\max}) \gg \max_j P(\mathcal{D}|\theta,\varepsilon^{(j)}), \tag{5}$$

e) Loglikelihood:

$$\log P(\mathcal{D}|\theta) = \sum_i [\log\mathcal{L}_i + \log(1-p_i)]\varepsilon_i^{\max} + [\log p_i - \log\Delta](1-\varepsilon_i^{\max}) \tag{6}$$

Computing the Posterior

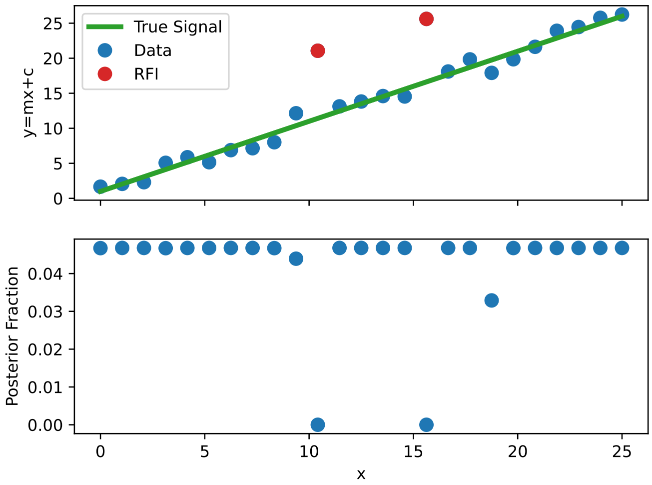
e) Loglikelihood:

$$\log P(\mathcal{D}|\theta) = \sum_i [\log \mathcal{L}_i + \log(1 - p_i)]\varepsilon_i^{\mathrm{max}} + [\log p_i - \log \Delta](1 - \varepsilon_i^{\mathrm{max}}) \tag{6}$$

f) Maximise ε^{max} by comparing the terms:

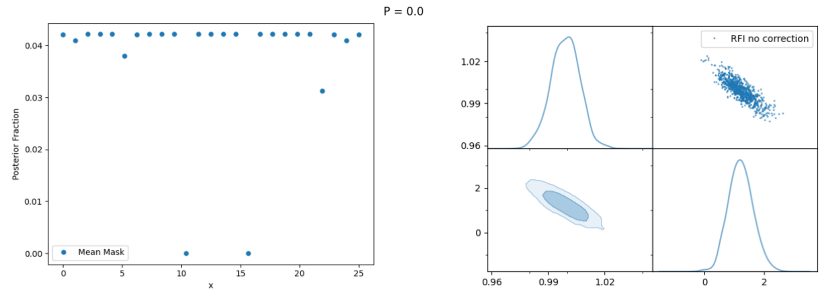
$$\log P(\mathcal{D}|\theta) = \begin{cases} \log \mathcal{L}_i + \log(1 - p_i), & \text{if } [\log \mathcal{L}_i + \log(1 - p_i) > \log p_i - \log \Delta] \\ \log p_i - \log \Delta, & \text{otherwise} \end{cases} \tag{7}$$

Fit on a simple toy model

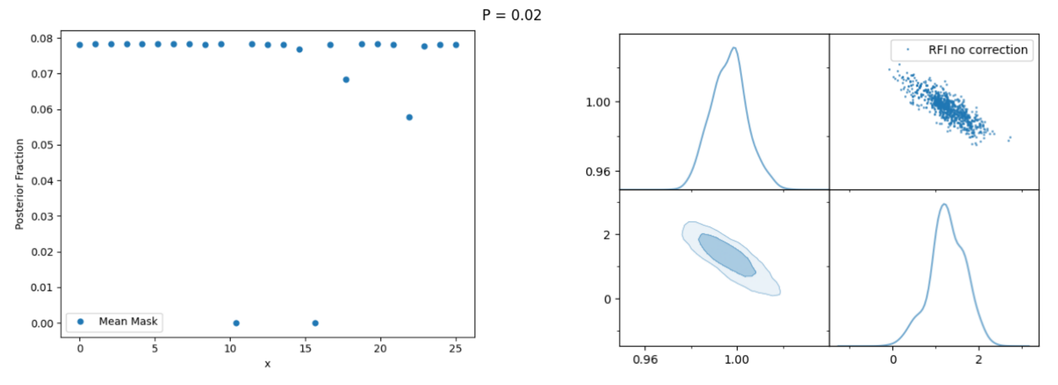


Likelihood thresholding condition p

$$\log P(\mathcal{D}|\theta) = \sum_i [\log \mathcal{L}_i + \log(1 - p_i)]\varepsilon^{\max} + [\log p_i - \log \Delta](1 - \varepsilon_i^{\max}) \tag{7}$$

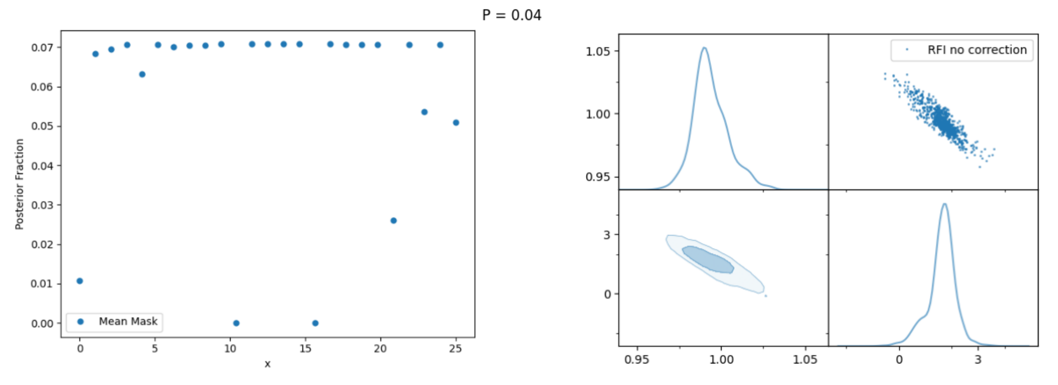


Varying p



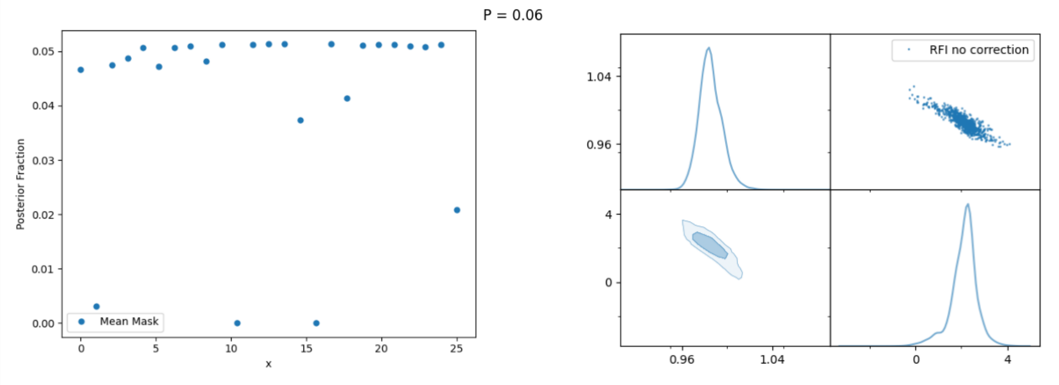
P = 0.02
Posterior Fraction
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0.00
Mean Mask
0
5
10
15
20
25
x
RFI no correction
1.00
0.96
2
0
0.96
1.00
0
2

Varying p



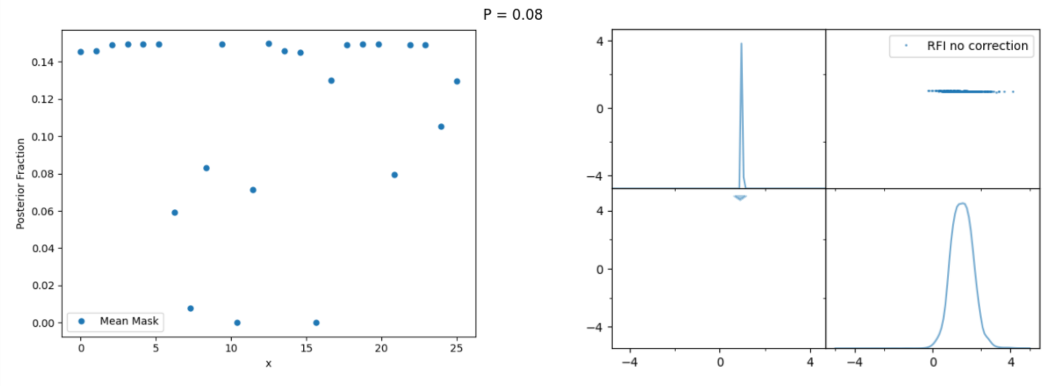
P = 0.04
Posterior Fraction
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0.00
Mean Mask
0
5
10
15
20
25
x
RFI no correction
1.05
1.00
0.95
3
0
0.95
1.00
1.05
0
3

Varying p

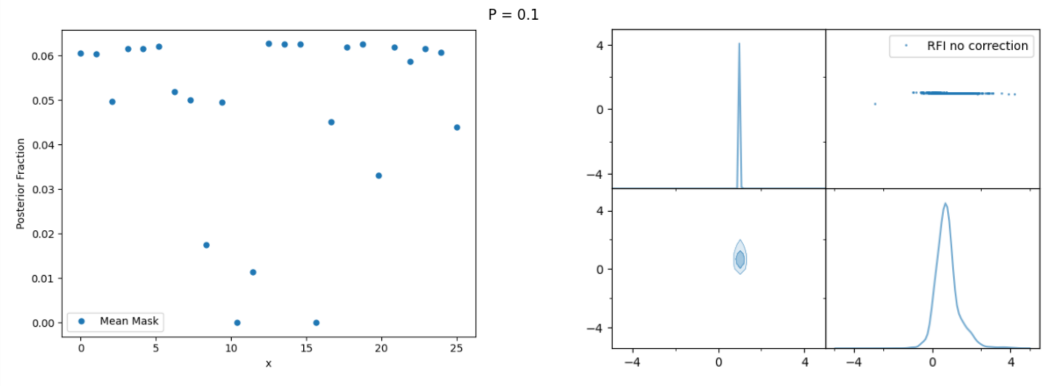


P = 0.06
Posterior Fraction
0.05
0.04
0.03
0.02
0.01
0.00
Mean Mask
0
5
10
15
20
25
x
RFI no correction
1.04
0.96
4
0
0.96
1.04
0
4

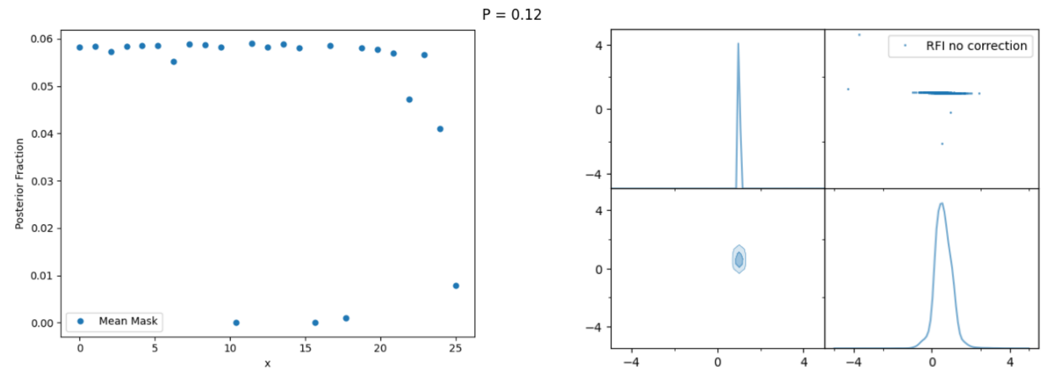
Varying p



Varying p

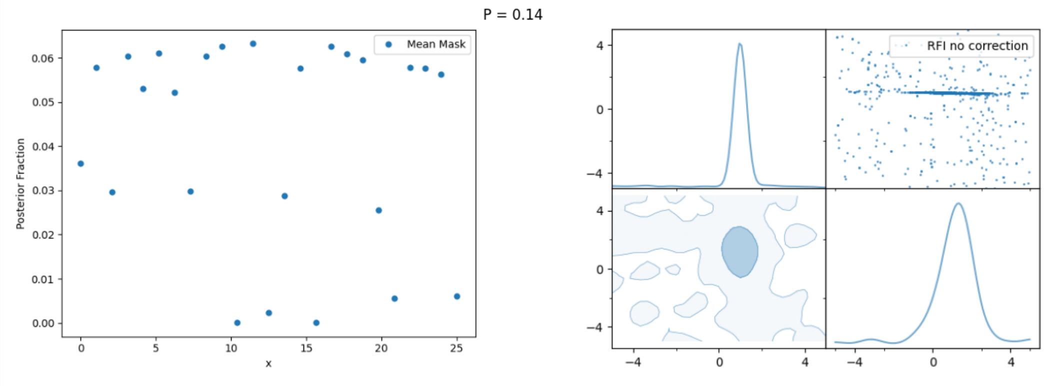


Varying p

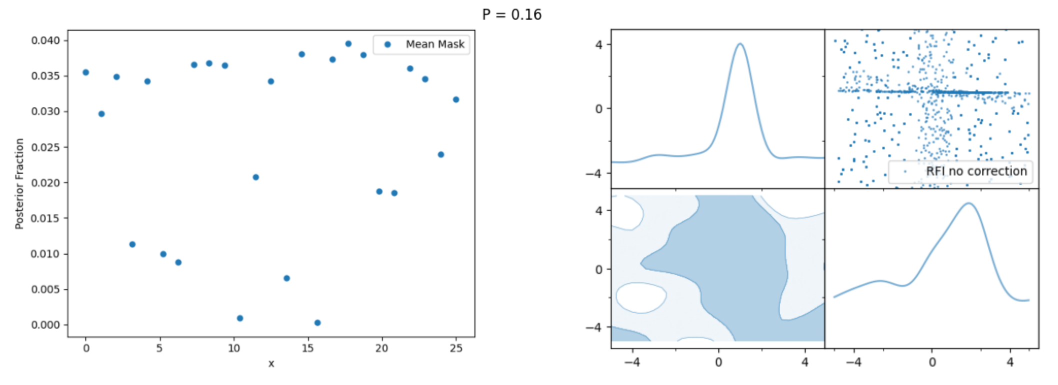


P = 0.12
Posterior Fraction
0.06
0.05
0.04
0.03
0.02
0.01
0.00
Mean Mask
0
5
10
15
20
25
x
RFI no correction
4
0
−4

Varying p


P = 0.14
Mean Mask
Posterior Fraction
x
0.00
0.01
0.02
0.03
0.04
0.05
0.06
0
5
10
15
20
25
RFI no correction
4
0
−4

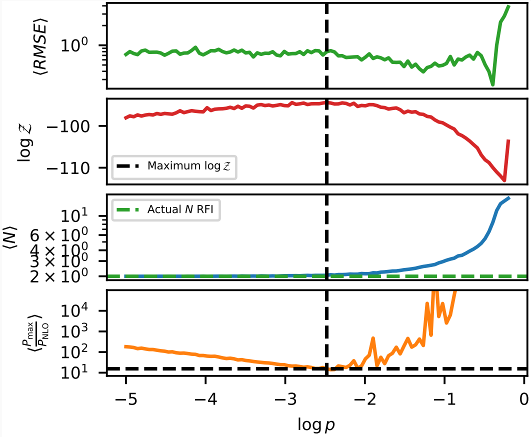
Varying p



P = 0.16
Mean Mask
Posterior Fraction
x
0.040
0.035
0.030
0.025
0.020
0.015
0.010
0.005
0.000
0
5
10
15
20
25
RFI no correction
4
0
−4

Selection strategy for p.

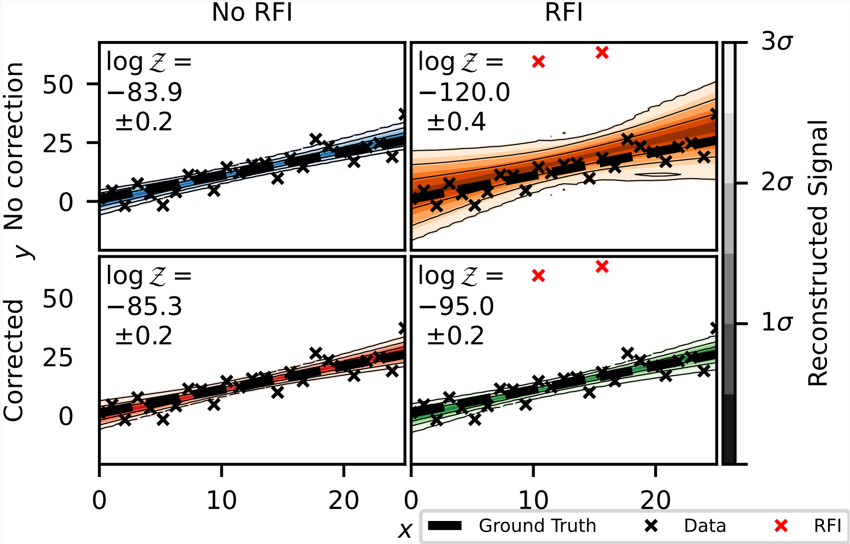
- 'Select p such that the Bayesian evidence is maximised'



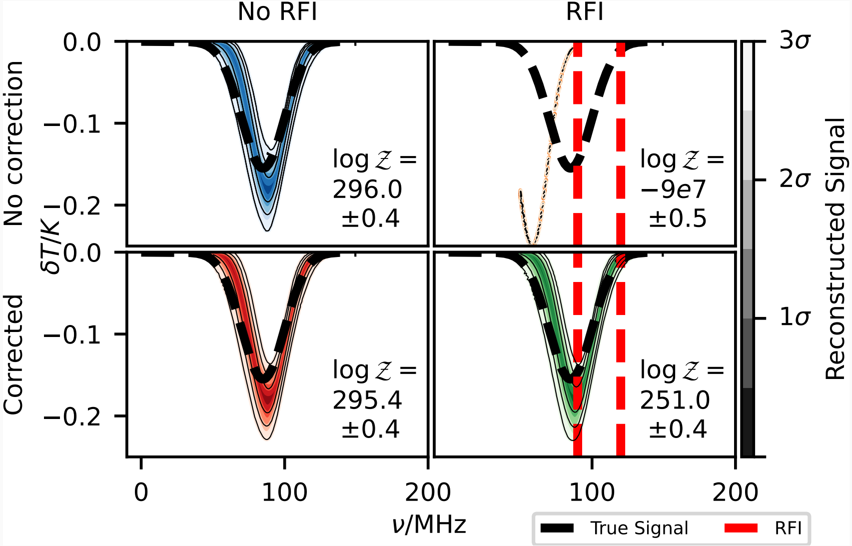
Fully automated anomaly detection

- Putting a prior on p, we can fit it dynamically as a free parameter.
- This fully automates the anomaly detection process.
- Must exclude $p = 0$.

Application to toy model



Application to REACH data



Implement with 2 lines of code

```python
def likelihood(theta):
    sig = theta[0]
    logL = -(f_noise - window)**2/sig**2/2 - np.log(2*np.pi*sig**2)/2
    return logL, []
```

```python
def likelihood(theta):
    sig = theta[0]
    logL = -(f_noise - window)**2/sig**2/2 - np.log(2*np.pi*sig**2)/2 + np.log(1-p)
    emax = logL > logp - np.log(delta)
    logPmax = np.where(emax, logL, logp - np.log(delta)).sum()
    return logPmax, []
```

Tutorial @ github.com/samleeney

Problems with this method

- Requires some knowledge of a real model for the data
    - Cannot be used as a blind search tool
    - Model must be reasonably accurate
- Currently struggling to detect RFI on REACH
    - Not at noise level for 21cm signal
    - Signal-to-noise ratio affects detection efficiency
- Implementation challenges
    - Requires existing Bayesian pipeline
    - Can be difficult to integrate into established projects
    - Computational overhead may be significant

Read the paper!

PHYSICAL REVIEW D **108,** 062006 (2023)

Bayesian approach to radio frequency interference mitigation

S. A. K. Leeney,[*,§] W. J. Handley,[†,§] and E. de Lera Acedo[‡,§]

The University of Cambridge, Astrophysics Group, Cavendish Laboratory,
J. J. Thomson Avenue, Cambridge, CB3 0HE, United Kingdom

(Received 5 May 2023; accepted 29 August 2023; published 29 September 2023)

Interfering signals such as radio frequency interference from ubiquitous satellite constellations are becoming an endemic problem in fields involving physical observations of the electromagnetic spectrum. To address this we propose a novel data cleaning methodology. Contamination is simultaneously flagged and managed at the likelihood level. It is modeled in a Bayesian fashion through a piecewise likelihood that is constrained by a Bernoulli prior distribution. The techniques described in this paper can be implemented with just a few lines of code.

DOI: 10.1103/PhysRevD.108.062006

arxiv: 2211.15448

A new use case: supernovae anomaly detection

SALT and SNooPy Models

SALT Model:

$$F(p,\lambda) = x_0\left[M_0(p,\lambda) + x_1 M_1(p,\lambda) + \ldots\right] \times \exp\left[c \times CL(\lambda)\right]$$

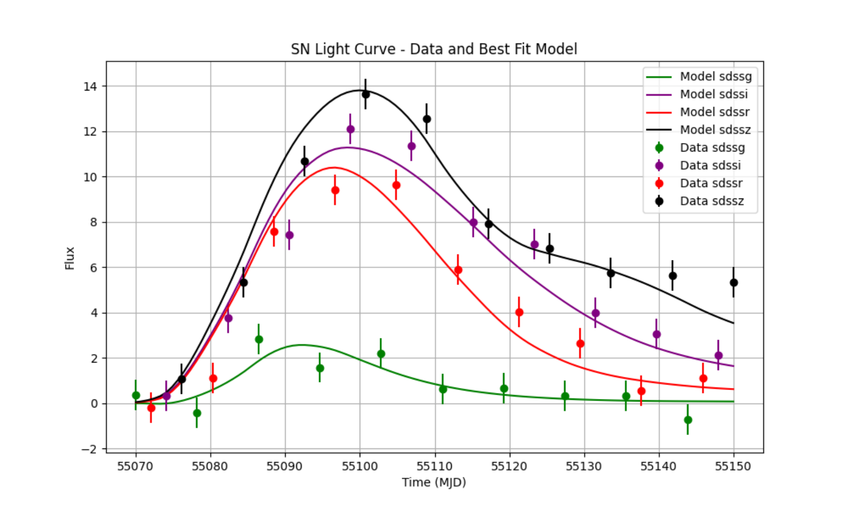
Where:

- Free parameters are: x_0, x_1, t_0, and c
- Model surface parameters are: $M_0(p,\lambda)$, $M_1(p,\lambda)$
- p is a function of redshift and t_0

SNooPy Model:

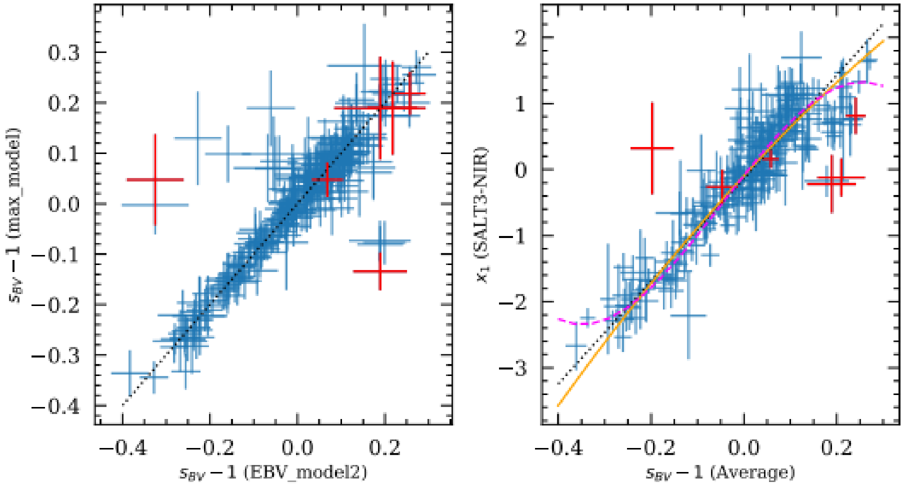
- Uses empirical templates with shape parameters (Δm_{15} or s_{BV})
- Models host galaxy extinction using reddening laws

Light Curve Image



Anomaly detection by comparing SALT and SNooPy

22 *A. Do et al.*



Time sensitive anomaly detection

Time sensitive likelihood

- Likelihood from before is extended into two dimensions, becoming

$$\log \mathcal{L}(\theta) = \sum_{ij} \left[\log \mathcal{L}_{ij}(\theta) + \log(1 - p_{ij})\right] \epsilon_{ij} + \left[\log p_{ij} - \log \Delta\right](1 - \epsilon_{ij}) \quad (8)$$

Time sensitive likelihood

- Computation time of the likelihood grows linearly with the number of time bins used in the data set.

$$\log \mathcal{L}(\theta) = \sum_{ij} \left[\log \mathcal{L}_{ij}(\theta) + \log\left(1 - p_{ij}\right)\right] \epsilon_{ij} + \left[\log p_{ij} - \log \Delta\right]\left(1 - \epsilon_{ij}\right) \quad (9)$$

Speeding up

- Anstey proposes a solution to this problem in Anstey et al. (2023)
- Common model per time bin is fit jointly.
- Speeds up fit but not sensitive to transients.

$$
\begin{aligned}
\log \mathcal{L} = & -\frac{1}{2} N_{\mathrm{t}} N_{\nu} \log(2\pi\sigma_{\mathrm{n}}) \\
& - \frac{D}{2{\sigma_{\mathrm{n}}}^2} - \frac{1}{2{\sigma_{\mathrm{n}}}^2} \Bigg[\sum_i \sum_k K_{\mathrm{sq}\,i\,k} F(\nu_i, \theta_{\mathrm{F}\,k})^2 \\
& + \sum_{k_1} \sum_{k_2} \Bigg[\sum_i K_{\mathrm{cross}\,i\,k_1\,k_2} F(\nu_i, \theta_{\mathrm{F}\,k_1}) F(\nu_i, \theta_{\mathrm{F}\,k_2}) \Bigg] \\
& - \mathrm{Tr} \sum_i K_{\mathrm{cross}\,i\,k_1\,k_2} F(\nu_i, \theta_{\mathrm{F}\,k_1}) F(\nu_i, \theta_{\mathrm{F}\,k_2}) \Bigg] \\
& - \frac{1}{{\sigma_{\mathrm{n}}}^2} \sum_i \Bigg[\frac{N_{\mathrm{t}}}{2} T_{\mathrm{S}}(\nu_i, \theta_{\mathrm{S}})^2 - \sum_k T_{\mathrm{D\cdot K}\,i\,k} F(\nu_i, \theta_{\mathrm{F}\,k}) \\
& - T_{\mathrm{D}\,i} T_{\mathrm{S}}(\nu_i, \theta_{\mathrm{S}}) + \sum_k K_k F(\nu_i, \theta_{\mathrm{F}\,k}) T_{\mathrm{S}}(\nu_i, \theta_{\mathrm{S}}) \Bigg] . \qquad (7)
\end{aligned}
$$

Likelihood reweighting

- Introduced in context of gravitational waves by Payne et al. (2019) and Romero-Shaw et al. (2019)
- Bayesian sampling techniques spend lots of time at tails of distribution.
- Reweighting is essentially a coarse then fine search, using a simple (fast) then complex (slow) likelihood.

$$p(\theta|d) = \frac{L(d|\theta)\pi(\theta)}{Z} = w(d|\theta)\frac{L_0(d|\theta)\pi(\theta)}{Z_0} \tag{10}$$

$$w(d|\theta) \equiv \frac{L(d|\theta)}{L_0(d|\theta)} \tag{11}$$

$$Z = Z_0 \int d\theta\, p_0(\theta|d) \left[\frac{L(d|\theta)}{L_0(d|\theta)}\right] \tag{12}$$

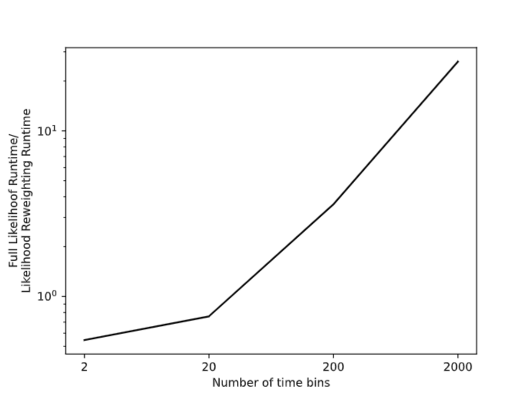
$$= Z_0 \sum_{k=1}^{n} w(d|\theta_k) \tag{13}$$

Carrying out Bayesian inference with the proposal likelihood, we obtain "proposal posterior samples" for the distribution

$$p_0(\theta|d) = \frac{L_0(d|\theta)\pi(\theta)}{Z_0}$$

How does this help us?

- We can use the fast method for a coarse scan.
- Then the slower method for refined scan.
- Increases speed massively for larger problems.



Read the paper!

Enhanced Bayesian RFI Mitigation and Transient Flagging Using Likelihood Reweighting

Dominic Anstey,[1,2★] Sam A. K. Leeney,[1,2]
[1]Astrophysics Group, Cavendish Laboratory, J. J. Thomson Avenue, Cambridge, CB3 0HE, UK
[2]Kavli Institute for Cosmology, Madingley Road, Cambridge, CB3 0HA, UK

Accepted XXX. Received YYY; in original form ZZZ

ABSTRACT
Contamination by Radio Frequency Interference (RFI) is a ubiquitous challenge for radio astronomy. In particular, transient RFI is difficult to detect and avoid, especially in large data sets with many time bins. In this work, we present a Bayesian methodology for time-dependent, transient anomaly mitigation. In general, the computation time for correcting for transient anomalies in time-separated data sets grows proportionally with the number of time bins. We demonstrate that utilising likelihood reweighting can allow our Bayesian anomaly mitigation method to be performed with a computation time close to independent of the number of time bins. In particular, we identify a factor of 25 improvement in computation time for a test case with 2000 time bins. We also demonstrate how this method enables the flagging threshold to be fit for as a free parameter, fully automating the mitigation process. We find that this threshold fitting also prevents overcorrecting of the data in the case of wide priors. Finally, we investigate the potential of the methodology as a transient detector. We demonstrate that the method is able to reliably flag an individual anomalous data point out of 302,000 provided the SNR > 10.

Key words: methods: data analysis – radio continuum: transients

arxiv: 2310.02146

Conclusions

- Anomaly detection in the likelihood domain
- Simultaneous anomaly detection and mitigation
- Implement with 2 lines of code!

References

Anstey, D., de Lera Acedo, E., and Handley, W. (2023). Use of time dependent data in bayesian global 21-cm foreground and signal modelling. *Monthly Notices of the Royal Astronomical Society*, 520(1):850–865.

Payne, E., Talbot, C., and Thrane, E. (2019). Higher order gravitational-wave modes with likelihood reweighting. *Physical Review D*, 100(12):123017.

Romero-Shaw, I. M., Lasky, P. D., and Thrane, E. (2019). Searching for eccentricity: signatures of dynamical formation in the first gravitational-wave transient catalogue of ligo and virgo. *Monthly Notices of the Royal Astronomical Society*, 490(4):5210–5216.